

Name - ~~Amar Singh~~
PRN - 1032234264
Name - ~~Amar Singh~~
Amar Singh

ESD Lab Assignment-01

Aim: To develop responsive web design using HTML5, containing a form. Style pages using CSS. Use tag selector, class selector & id selectors. Use inline, internal & external css. Apply bootstrap CSS.

Objectives:

- i) Understand HTML tags
- ii) Learn styling of web pages using CSS.
- iii) Learn bootstrap front end framework.

Problem statement: Create single page portfolio showing casing your skills & projects. Page must include sections about me & contact. Design must be responsive, employing bootstrap's system. Styling should be implemented using mix of inline, internal & external css.

Theory:

① Define ~~responsive~~ web design (RWD). What is its primary goal?

→ ~~Responsive~~ web design is a web design approach that ~~assures~~ ensures website looks & functions well on all devices. Its primary goal is to create a flexible, user friendly interface that automatically adapts layout, images & other elements based on device's screen size.

- ② Explain role of the `<meta name="viewport">` tag. Why is it essential for RWD?
- The tag tells browser how to control page dimension & scaling.
`width=device-width` → width matches device screen width.
`initial-scale=1.0` → sets initial zoom = 100%.
Without it mobile may shrink or zoom out pages.
- ③ How does bootstrap assist in creating responsive layout? discuss concept of grid system & how it adapts to different screen size.
- Bootstrap helps create responsive layouts by providing 12 column grid system. You place content inside columns & bootstrap automatically arranges them.

- ④ Differentiate btwn tag, class, ID selectors.

	Syntax	Applies to	Uniqueness
Tag selector	<code>p {color: blue;}</code>	All <code>p</code> elements	Not unique
Class selector	<code>.highlight {color: red;}</code>	All elements with <code>class = "highlight"</code>	Can be reused
ID selector	<code>#main {background: yellow;}</code>	Single element with <code>id = "main"</code>	Must be unique

⑤ Describe three main ways to apply css to HTML.

i) Inline css

→ Applied directly to an element using style attribute

ii) Internal css.

Written inside a <style> tag in HTML <head>

iii) External css

stored in separate css file & linked using
<link rel="stylesheet" href="style.css">

32
218/26

Assignment-02

Aim: Develop a web application using javascript to implement sessions, cookies, DOM. Perform validations such as checking for emptiness, only numbers for phone numbers, special character requirement for password, regular expressions for certain format of the fields etc.. Use the MySQL database.

Objectives:

- 1) To understand what form validation is.
- 2) To learn basic functioning of DOM objects.
- 3) To learn how to apply various techniques to implement it.

Theory:

1) Role of Regular Expressions (RegEx)

RegEx are patterns for matching strings, useful for validating formats like phone numbers, passwords, or emails by enforcing rules on characters.

2) Explain the fundamental difference between a session and a cookie in the context of web application development. How do they work together to maintain a user's logged-in state?

→ Difference between session and cookie.

- Cookies are small pieces of data stored on the client-side used to remember user preferences or identifiers.
- Sessions store user data on the server, identified by a session ID usually stored in a cookie. Together, they maintain user login states securely by keeping sensitive data on the server-side (sessions) while using cookies for identification.

- 3) What is the purpose of performing both client-side and server-side validation? Describe a scenario where relying solely on client-side validation could lead to a security vulnerability.
- Client-side validation provides instant feedback and reduces server load.
- Server-side validation ensures security by validating data regardless of client manipulation. Relying only on client-side validation can be bypassed by attackers, leading to security risks such as SQL injection or data corruption.

- 4) Provide a simple example of how a JavaScript script can interact with the DOM to dynamically change the content of a web page after a user action, such as a form submission.

→ `<p id="msg">Hello</p>`
`<button onclick="changeText()">Click Me</button>`

```
<script>
    function changeText() {
        document.getElementById("msg").innerHTML
            = "Text changed!";
    }
</script>
```

- ① Define response
a=?

the purpose of performing both client-side and server-

client-side
load.

-side

etc of
an e-
ch as

a s-
t w/
pa-
id-
on

pt>
actic
doc

3
/scr

..
F...
.

...

...

...

- 5) Give the steps for connectivity from Frontend using HTML CSS JS to MySQL.

→ Steps for frontend to MySQL connectivity.

- Create HTML form to collect data.
- Use JavaScript/AJAX to send data to server.
- Server-side script (Node.js, PHP, etc.) receives data.
- Server-side script connects to MySQL databases.
- Insert or query data in MySQL.
- Send response back to frontend.

FAQ's

- Q.1 Write 3 reasons why Form validations are important.

→ Three reasons why form validations are important.

- Prevent submission of incomplete or incorrect data.
- Improve user experience with instant feedback.
- Enhance security by avoiding malicious inputs.

- Q.2 Give an example of how to modify an attribute value using DOM.

→ `document.getElementById('myImage').setAttribute('src', 'newImage.jpg');`

- Q.3 What are the different features of Java Script?

→ Client-side scripting language.

• Dynamic typing

• Event-driven programming.

• DOM manipulation capability.

• Supports object-oriented and functional programming.

Conclusion:

This assignment demonstrates the importance of client-side form validation, efficient DOM manipulation using JS, and integration of jQuery for enhanced interactivity.

~~Integrating client-side validation and server-side validation is a good practice for better user experience. It helps to provide instant feedback to the user and reduces the load on the server by catching errors early. The validation rules can be defined using regular expressions or external validation libraries like Validator.js. It's also important to validate both client-side and server-side to ensure consistency and security.~~

~~The validation logic can be implemented using JavaScript or jQuery. It's recommended to use a library like jQuery Validation to handle complex validation rules and reduce code duplication.~~

Assignment - 03

Aim: Design an interactive front-end application using React by implementing template using components, states & props, class, events. It must be responsive to scale across different platforms.

Objective: To develop a responsive, interactive front-end application using React.js that effectively demonstrates component-based architecture, state management, & event handling with scalable UI components, dynamic data via states & props, & seamless user interactions across devices.

Theory:

- 1) Explain the role of state & props in React. How do they differ, & what is the primary purpose of each in managing data flow within a component-based application.
- State: Represents mutable data owned & managed by a component. It allows components to create dynamic & interactive UI's by updating themselves when the state changes.

Props: Short for 'properties', props are read-only inputs passed from a parent component to a child, enabling data sharing across components.

Difference: State is internal & changeable, while props are external & immutable. Together, they enable unidirectional data flow in react.

2) What is a React component? Differentiate between a class component & a functional components & discuss the advantages of using a functional component with hooks like useState & useEffect over a class component.

→ Component :- A reusable, independent piece of UI in React.
 Class Components :- Defined using ES6 classes, use this state & lifecycle methods.

Functional components :- Defined as functions, use hooks like useState & useEffect for state & lifecycle management.

Advantages of functional + hooks :- Cleaner syntax, less boilerplate, better performance, easier to test, & modern React development favors hook over classes.

3) Describe the concept of "templating using components" in React. Why is this approach considered superior to traditional web development methods that rely on monolithic HTML files?

→ Breaks UI into reusable, modular components.
 → Superior to monolithic HTML → improves reusability, scalability & maintainability.

4) How do you handle user events in React? Provide a simple code snippet to demonstrate how an event handler is defined in a component how it can be used to update the component state.

→

→ Handlers defined as functions; update state using `useState`

```
import React {useState} from "react";
function Counter() {
  const [count, setCount] = useState(0);
  return (
    <div>
      <p>{count}</p>
      <button onClick={() => setCount(count + 1)}>Inc</button>
    </div>
  );
}
```

5) What is responsive design, why is it crucial for modern application? Describe how you would implement a responsive design in a React application using CSS media queries or a CSS in JS library.

→ Ensures UI adopts to all screen sizes/devices

Implement using CSS media queries or CSS in JS

Eg:- `@media (max-width: 600px) {`
 `div { font-size: 10px; }`
`}`

(20)
20 apps

Assignment - 4

Aim: Enhance web page developed in earlier assignment by rendering Lists and Portals, Error Handling, Portals and style with React CSS also make it a responsive design to scale well across PC, tablet and mobile phone.

Objectives:

- Enhance User Interface and Experience.
- Improve Application Robustness and Navigation.

Theory:

Q.1) How do Lists and keys work in React?

- Ans → Lists in React allow you to render multiple elements dynamically using Javascript's map() method to loop over data (e.g. an array of objects).
- Keys are unique identifiers assigned to each element in a list to help React efficiently update and render only the changed items during re-renders.

Q.2) What is a React Portal and when would you use one?

Ans • React Portal is a way to render children outside their parent component's DOM hierarchy.

- It is useful when you need to render components that are visually or functionally separate, like modals, tool tips or pop-ups, but want to keep them logically within the same component structure.

Q.3) Discuss the importance of error boundaries in React.

Ans Error boundaries are components that catch Javascript errors anywhere in their child component tree, log those errors and then finally display a fall back UI.

They prevent the entire application from crashing when an error occurs, providing a more user-friendly experience by showing a graceful fall back, instead of a broken UI.

Q.4) How does React Router enable single page application (SPA) functionality?

Ans React Router enables SPA functionality by allowing you to define routes and navigate between different components without causing full-page reloads. It dynamically changes the view in response to URL changes, providing a seamless user experience within client-side routing.

Q.5) Explain the different ways to style a React application.

Ans Inline Styling :- Using the style attribute with objects.

CSS style sheets :- Importing regular .css files & applying class names.

CSS Modules :- Scoped styles using .module.css files to avoid naming conflicts.

Conclusion: This assignment has my React knowledge by adding lists, portals, error handling, routing and responsive styling, enhancing usability, navigation and reliability across devices.

✓
M
29/9/15

Assignment - 05

Aim: Develop a responsive web design using Express Framework to perform CRUD operations & deploy with Node.js use MongoDB.

Objectives

- 1) Develop a full-stack web Application.
- 2) Demonstrate Backend Development & Deployment Proficiency.

Theory:

- Q.1) What is the role of Express.js as a web framework for Node.js?
- Provides a lightweight, fast & flexible web framework built on top of Node.js.
- Simplifies handling HTTP requests & responses.
 - Supports routing.
 - Middleware support for request processing.
 - Enables building RESTful APIs & full-stack applications efficiently.
 - Reduces boilerplate code compared to using pure Node.js.

Q.2) What is a React Portal & when would you use one?

→ A portal allows you to render a component's children into a DOM node that exists outside the parent component's hierarchy.

Syntax: `ReactDOM.createPortal(child, container)`

Useful for:

- Avoiding CSS overflow/positioning issues.

- Models, dialogs, tooltips, dropdowns.

Q.3) Discuss the importance of Error Boundaries in React.

→ Error Boundaries are React components that catch JavaScript errors in their child component tree.

- They prevent the entire app from crashing when an error occurs.

- They can show a fallback instead of breaking the whole app.

- Crucial for production apps where stability & user experience matter.

Q.4) How does React Router enable Single Page Application (SPA) functionality?

→ React Router manages navigation without reloading the page.

- It uses the History API to update the URL while keeping the app alive.

- Components are rendered conditionally based on the route, enabling seamless transitions.

- Q.S) Explain the different ways to style a React application.
- Inline Styling → Using the style attribute with objects.
 - CSS stylesheet → Importing regular.css files & applying class names.
 - CSS modules → Scoped styles using .module.css files to avoid naming conflicts.

Conclusion: In summary, express JS simplifies backend development for node.js, while mongoDB offers flexible, scalable database solⁿ. This helps to demonstrate both backend & deployment skills.

By
11/10/25